

```
In [11]: import numpy as np
import math as m
from math import sin,cos
import pandas as pd
from matplotlib import pyplot as plt
```

```
In [2]: def g(x):
df=2*x-4*sin(x)
ddf=2-4*cos(x)
return x-df/ddf
```

```
In [3]: f=lambda x:x**2+4*cos(x)
```

```
In [4]: def Newton(x_0,tol):
x=np.array([x_0])
while abs(g(x[-1])-x[-1])>tol:
x=np.append(x,g(x[-1]))
return x
```

```
In [5]: arr=Newton(1,10e-20)
```

```
In [6]: vf=np.vectorize(f)
```

```
In [42]: df=pd.DataFrame({'x^{(k)}$:arr,'$f(x^{(k)})$:vf(arr),'$\epsilon$:np.insert(a
df.index.name="iteration number"
```

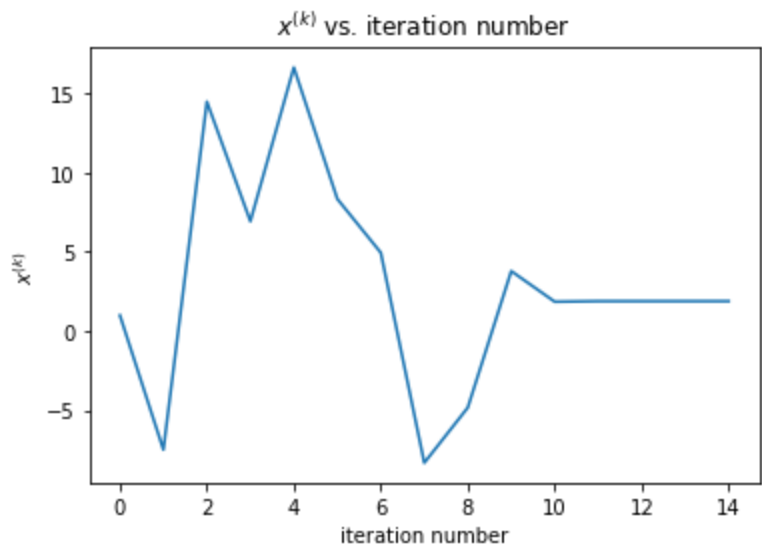
```
In [43]: df
```

Out[43]:

	$x^{(k)}$	$f(x^{(k)})$	ϵ
iteration number			
0	1.000000	3.161209	NaN
1	-7.472741	57.330143	8.472741e+00
2	14.478521	208.288517	2.195126e+01
3	6.935115	51.275483	7.543406e+00
4	16.635684	274.347348	9.700569e+00
5	8.343938	67.738946	8.291747e+00
6	4.954633	25.507911	3.389305e+00
7	-8.301318	67.181618	1.325595e+01
8	-4.817320	23.625525	3.483998e+00
9	3.792574	11.201663	8.609894e+00
10	1.861061	2.318725	1.931513e+00
11	1.896214	2.316809	3.515312e-02
12	1.895495	2.316808	7.196385e-04
13	1.895494	2.316808	2.995885e-07
14	1.895494	2.316808	5.195844e-14

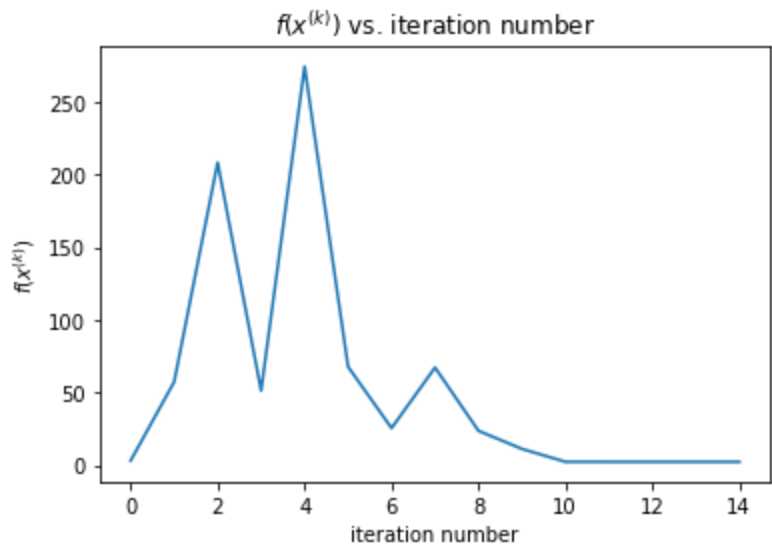
```
In [32]: plt.plot(df.index,df.iloc[:,0])
plt.xlabel("iteration number")
plt.ylabel("$x^{(k)}$")
plt.title("$x^{(k)}$ vs. iteration number")
```

```
Out[32]: Text(0.5, 1.0, '$x^{(k)}$ vs. iteration number')
```



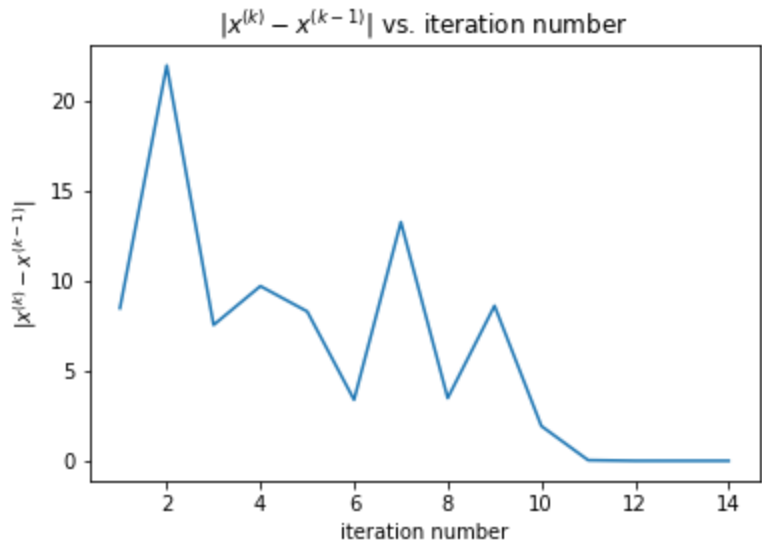
```
In [33]: plt.plot(df.index,df.iloc[:,1])
plt.xlabel("iteration number")
plt.ylabel("$f(x^{(k)})$")
plt.title("$f(x^{(k)})$ vs. iteration number")
```

Out[33]: Text(0.5, 1.0, '\$f(x^{(k)})\$ vs. iteration number')



```
In [44]: plt.plot(df.index,df.iloc[:,2])
plt.xlabel("iteration number")
plt.ylabel("$|x^{(k)}-x^{(k-1)}|$")
plt.title("$|x^{(k)}-x^{(k-1)}|$ vs. iteration number")
```

Out[44]: Text(0.5, 1.0, '\$|x^{(k)}-x^{(k-1)}|\$ vs. iteration number')



```
In [ ]:
```